

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-325197

(43)Date of publication of application : 25.11.1994

(51)Int.Cl.

G06K 7/10

(21)Application number : 05-136910

(71)Applicant : FUJITSU LTD

(22)Date of filing : 14.05.1993

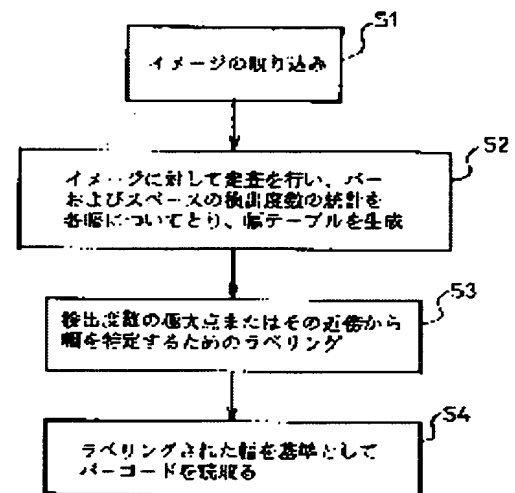
(72)Inventor : MORI YUICHIRO

(54) BAR CODE RECOGNIZING METHOD

(57)Abstract:

PURPOSE: To heighten the recognition rate of a bar code and to eliminate the need to previously designated module width.

CONSTITUTION: The bar code is fetched in as an image, which is scanned; and the detection frequency of the widths of the bars and spaces of the bar code are statistically processed through the scanning to generate a width table. Further, labeling which specifies bars and spaces from the point of the maximum detection frequency or its vicinity is performed by referring to the width table and the bar code is read and recognized on the basis of the specified width.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

BEST AVAILABLE COPY

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平6-325197

(43)公開日 平成6年(1994)11月25日

(51)IntCl.⁵

G 0 6 K 7/10

識別記号

片内整理番号

F I

技術表示箇所

V 9191-5L

審査請求 未請求 請求項の数6 F D (全 14 頁)

(21)出願番号 特願平5-136910

(22)出願日 平成5年(1993)5月14日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72)発明者 森 雄一郎

神奈川県横浜市港北区新横浜2丁目4番19

号 株式会社富士通プログラム技研内

(74)代理人 弁理士 宇井 正一 (外4名)

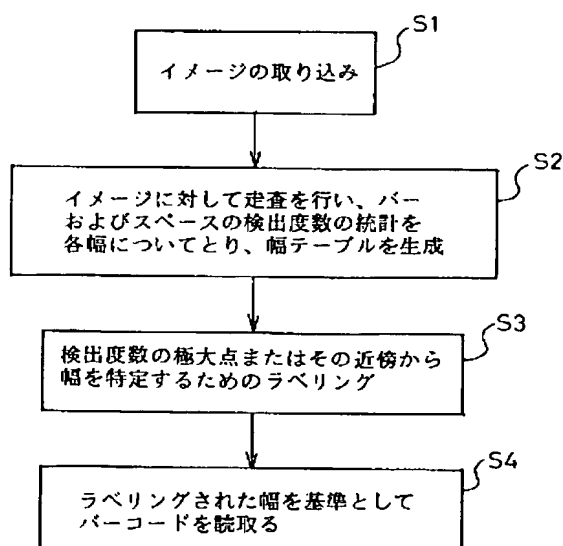
(54)【発明の名称】 バーコード認識方法

(57)【要約】

【目的】 バーコードの認識方法に関し、バーコードの認識率を高めると共に、モジュール幅を予め指定することを不要とすることを目的とする。

【構成】 バーコードをイメージとして取り込み、そのイメージを走査し、走査を通じてバーコードのバーの幅とスペースの幅の各々の検出度数について統計を取り、幅テーブルを生成する。さらにこの幅テーブルを参照して、検出度数の極大点またはその近傍よりバーおよびスペースを特定するラベリングを行い、ここに特定された幅を基準として、バーコードの読取りならびに認識を行うように構成する。

本発明に係る方法の基本ステップを表すフローチャート



【特許請求の範囲】

【請求項 1】 バーコードを構成する複数のバーおよびスペースの各パターンを、イメージとして取り込む第 1 ステップと、

前記イメージに対して走査を行い、前記複数のバーおよびスペースの各々の幅を検出すると共に、その検出度数の統計を該幅の各々についてとり、前記バーおよびスペースの各々について幅テーブルを生成する第 2 ステップと、

前記幅テーブルを参照し、少なくとも 2 つの前記検出度数の極大点またはその近傍から前記幅を特定するラベリングを前記バーおよびスペースの各々について行う第 3 ステップと、

前記バーおよびスペースの各々について少なくとも 2 つのラベリングされた前記幅を基準にして前記バーコードの前記バーおよびスペースの読み取りを行う第 4 ステップ、とからなることを特徴とするバーコード認識方法。

【請求項 2】 バーコードを構成する複数のバーまたはスペースの各パターンを、イメージとして取り込む第 1 ステップと、

前記イメージに対して走査を行い、前記複数のバーまたはスペースの各々の幅を検出すると共に、その検出度数の統計を該幅の各々についてとり、前記バーまたはスペースについて幅テーブルを生成する第 2 ステップと、

前記幅テーブルを参照し、少なくとも 2 つの前記検出度数の極大点またはその近傍から前記幅を特定するラベリングを前記バーまたはスペースについて行う第 3 ステップと、

前記バーまたはスペースについて少なくとも 2 つのラベリングされた前記幅を基準にして前記バーコードの前記バーまたはスペースの読み取りを行う第 4 ステップ、とからなることを特徴とするバーコード認識方法。

【請求項 3】 前記第 2 ステップにおいて、前記イメージに対して複数回、ラスタの位置を変えながら走査を行い、各ラスタ毎に前記幅テーブルを生成する請求項 1 または 2 に記載のバーコード認識方法。

【請求項 4】 前記第 3 ステップにおいて、前記バーに関し、前記検出度数の合計度数に対する一定の割合を設定し、その設定度数を超える範囲の前記幅をもって前記のラベリングを行い、同様に、前記スペースに関し、前記検出度数の合計度数に対する一定の割合を設定し、その設定度数を超える範囲の前記幅をもって前記のラベリングを行う請求項 1 または 2 に記載のバーコード認識方法。

【請求項 5】 前記第 2 ステップにおいて、前記バーまたはスペースに関するそれぞれの前記検出度数の合計が、該バーまたはスペースのそれぞれについての既知の一定本数と一致したことを条件に、前記の統計をとり前記幅テーブルを生成する請求項 1 または 2 に記載のバーコード認識方法。

【請求項 6】 前記第 3 ステップにおいて、前記幅テーブルを参照し、前記幅を変数とする前記検出度数の曲線から該検出度数の上昇、下降および横這いを算出して、該曲線の極大および極小を割り出し、該極大を中心とし、かつ、該極小により区分された前記幅の範囲を切り出して、該幅を特定する前記のラベリングを行う請求項 1 または 2 に記載のバーコード認識方法。

【発明の詳細な説明】**【0001】**

【産業上の利用分野】 本発明はバーコードの認識方法に関する。バーコード読取装置においては、複数種類のモジュール幅を有するバーならびにスペースの組み合わせからなるバーコードイメージを、イメージスキャナによって取り込み、取り込んだバーコードイメージよりバーコードのパターン（デジタル情報）を認識する。

【0002】

【従来の技術】 認識の対象となるバーコードには一般に複数種類のモジュール幅がある。また、バーコードが印刷される領域も対象毎に区々である。したがって、全てのバーコードを、定められた 1 つのイメージスキャナで読み取することはできない。このため従来は次の 2 つの手法を採用してあらゆるバーコードに対応できるようにしている。

【0003】 (i) 第 1 の手法は、イメージスキャナより取り込んだバーコードイメージを一旦拡大または縮小処理することによって統一されたモジュール幅に規格化し、その後本来のバーコードイメージの読取りならびにバーコードの認識を開始する、というものである。

(ii) 第 2 の手法は、認識しようとするバーコードの各々についてそのバーコードのモジュール幅を予めバーコード読取装置に対して指示し、その後、指示されたモジュール幅を基準として、本来のバーコードイメージの読取りならびにバーコードの認識を開始する、というものである。

【0004】

【発明が解決しようとする課題】 上記従来の手法にはそれぞれ問題がある。

(i) 上記第 1 の手法についてみると、上記の拡大または縮小処理によって、取り込んだバーコードイメージが荒れてしまう場合が多い。この結果、バーコードの認識率が低下するという問題が生ずる。

【0005】 (ii) 上記第 2 の手法についてみると、バーコードを認識する前にまずそのバーコードの属性を調べモジュール幅を確認した上でバーコード読取装置にそのモジュール幅を指示する、という煩雑な操作を必要とし、認識対象となるバーコードの種類が多数混在する場合には極めて能率が悪くなるという問題が生ずる。したがって本発明は上記の諸問題点に鑑み、取り込んだバーコードイメージに拡大または縮小処理を施すことによる認識率の低下をなくし、あるいはバーコードの各々につ

いて予めそのモジュール幅を指示するという操作を不要とすることができる、バーコード認識方法を提供することを目的とするものである。

【0006】

【課題を解決するための手段】図1は本発明に係る方法の基本ステップを表すフローチャートである。図に示すとおり、本発明の方法は基本的に4つのステップS1～S4から構成される。

第1ステップS1：バーコードを構成する複数のバーおよびスペースの各パターンを、イメージとして取り込む。

【0007】第2ステップS2：前記イメージに対して走査を行い、前記複数のバーおよびスペースの各々の幅を検出すると共に、その検出度数の統計を、該幅の各々について、とり前記バーおよびスペースの各々について幅テーブルを生成する。

第3ステップS3：前記幅テーブルを参照し、少なくとも2つの前記検出度数の極大点またはその近傍から前記幅を特定するラベリングを前記バーおよびスペースの各々について行う。

【0008】第4ステップS4：前記バーおよびスペースの各々について少なくとも2つのラベリングされた前記幅を基準にして前記バーコードの前記バーおよびスペースの読み取りを行う。

【0009】

【作用】バーコードの認識を行うに際し、まずそのバーコード（これから認識しようとするバーコード）についてのバーおよびスペースの各幅を自動的に割り出し、割り出したその幅を基準として各バーおよびスペースを判断することによりバーコードの認識を行う。

【0010】

【実施例】図2はバーコードパターンの周知の一例を表す図である。図中、BPはバーパターン、特に幅の太いバーの部分を示し、bpは幅が細い方のバーを示す。またこれらのバーに挟まれたSPは幅の太いスペース、spは細い幅のスペースを表す。なお、図では、太細2種の幅からなるバーおよびスペースを示したが、この幅は3種以上になることもある。また図では一般によく用いられる、太：細＝3：1の比率を有するバーコードパターンを示す。

【0011】図2のバーコードパターンは通常イメージスキャナによって一旦、二次元のバーコードイメージとして装置内のメモリ（図示せず）に取り込まれる。このバーコードイメージもやはり図2のようなバーの黒ドットを“1”、スペースの白ドットを“0”としたデータパターンとしてそのメモリに格納される。次にバーコードの読み取りのためにラスタ走査を行う。各ラスタはR1、R2およびR3として3本例示されているが、ラスタ領域Rはバーの長さ方向に長く伸びており、ラスタ本数は実際には多数本に及ぶ。

【0012】図3は幅テーブルを作成するための統計処理を表す図である。図2のラスタ走査によって各幅毎の検出度数を集計する。細い幅のバーbpは多数回のラスタ走査により集計した結果、図3の曲線bpのように幅（WD）方向に分布する。他方、太い幅のバーBPは多数回のラスタ走査により集計した結果、図3の曲線BPのように幅（WD）方向に分布する。なお右端のTは全検出度数を合計した値を示す。

【0013】これらの検出度数の分布状況の統計は幅テーブルとして保持される。なお、図3では太い幅のバーBPと細い幅のバーbpについて図解したが、全く同様のことはスペースについても行われる（ただし、図3以降では主としてバーのみを代表して説明する）。つまり図3に関しては、太い幅のスペースSPと細い幅のスペースspについての各幅の分布が上記の多数回のラスタ走査で集計される。

【0014】結局、図3の曲線bpおよびBPのそれぞれの極大点（山）を中心として、細る方向と太る方向に分布が広がる。これはバーコード印刷の際のずれや量子化誤差に起因する。図4は各幅の細太を特定するラベリングを説明する図である。図3の統計結果に基づきラベリング（細太の分類）を行う。図4においてONと付した領域bpは、この領域内に入る幅を有するバーは細いバーであると判定する領域として特定されたことを意味する。また、ONと付した領域BPは、この領域内に入る幅を有するバーは太いバーであると判定する領域として特定されたことを意味する。それ以外の領域（OFF）に入る幅を有するバーはノイズ等として無視される。

【0015】上述した図3の分布はかなり理想的な場合であり、実際には2つの分布の隣り合う裾の部分が互いに重なってしまうことが多い。図5は隣接する分布が重なる場合の統計処理を説明する図である。2つの分布bpおよびBPのそれぞれの裾の部分が重なり、これらの間に第3の分布Bbpが生じてしまう。

【0016】このような図5に示す分布（bp、Bbp、BP）になると、図4に示すような明確なラベリングが困難になる。そこでこのような場合には、前記第3ステップS3（図1）において、バーBP、bpに関し、検出度数の合計度数（T）に対する一定の割合を設定し、その設定度数を超える範囲の幅（WD）をもって前記のラベリングを行い、同様に、スペースSP、spに関し、検出度数の合計度数に対する一定の割合を設定し、その設定度数を超える範囲の幅（WD）をもって前記のラベリングを行う。

【0017】図5を参照すると、合計検出度数Tに対する一定の割合（例えば3%）を図中のtとして設定し、この設定度数tを超える範囲の幅（図5のWb およびW_B）をもって前述したラベリングを行う。このことはスペースの度数分布に関しても全く同様である。図6は幅

テーブルを作成するための統計処理を具体例で示す図であり、図3に対応している。図3の中にさらに幅の要素11を描き加えたものである。また図3の右端に示す合計度数Tは、図6の積算カウンタ12によって計数される。

【0018】図6では説明上、幅の要素11を粗くとっているが、実際にはもっと細かくとる。図の例によれば、例えば1mmの幅の要素について複数回のラスタ走査の後、28回の検出が行われ、また1.1mmの幅の要素については同様に26回の検出が行われたことを示す。いずれかの幅の要素11に1が加算されるとき、同時に積算カウンタ12にも1を加算する。

【0019】どのバーがどの幅の要素に属するかを決定するに際しては、各ラスタ毎に黒ドットの連続と白ドットの連続とを区別する必要がある。図7は図2における任意の1つのラスタにおけるバーコードイメージの一部を示す図であり、1つの箱が1つのドットに対応し、黒のドットBLと白のドットWHとからなる。最も左端に位置する黒ドットをD₁とし、引続きD₂、D₃…とする。D₁がラスタ上のどの位置にあるかを示すX座標も別途定義される。

【0020】図8は幅テーブルを作成する工程を示すフローチャート（その1）、図9は幅テーブルを作成する工程を示すフローチャート（その2）である。なお、100番台の数字は各ステップを区別するための参照番号である。

ステップ101：複数本のラスタを走査するための、ラスタカウンタ（図示せず）を0で初期化する。このラスタカウンタは、バーコードイメージ領域内のラスタ番号（図2のR1、R2…）を表す。また、バーおよびスペースのそれぞれの合計本数のカウンタ（B、S）を0で初期化する。

【0021】ステップ102：ラスタを、最も左端にある黒点をD₁とするドット列D_n（図7）と考え、D₀を1で初期化する。また、ドット列D_nのnを特定するX座標の変数Xを1で初期化する。更に、幅テーブルの要素番号Wを1で初期化する。

ステップ103：ドット列内の点D_XとD_{X-1}を比較して異なっている（黒→白、白→黒の変化点である）ならば、ステップ104へ進む。同じならば、ステップ108へ進む。

【0022】ステップ104：ドット列内の点D_XとD_{X-1}が異なっているならば、D_{X-1}がバーまたはスペースの端点なので、そのどちらか判断をする（黒ならばYes、白ならばNo）。D_Xが0ならば、105へ進む。D_Xが1ならば、106へ進む。ただし、黒を“1”、白を“0”とする。

ステップ105：D_{X-1}までがバーなので、バーの幅テーブル（TB）の内、現在カウント中の幅（WD）の要素番号W（図6の11に相当）の要素TS_Wをインクリ

メントする。Wは要素番号である。また、バーを新たに検出したので、バーの合計本数のカウンタ（B）（図6の12に相当）をインクリメントする。

【0023】ステップ106：D_{X-1}までがスペースなので、スペースの幅テーブル（TS）の内、現在カウント中の幅（WD）の要素番号Wの要素TS_Wをインクリメントする。また、スペースを新たに検出したので、スペースの合計本数のカウンタ（S）（図6の12に相当）をインクリメントする。

ステップ107：バーまたはスペースの検出をしたので、次のスペースまたはバーの幅をカウントするために、各要素11に対応して設けられる幅テーブルカウンタ（W）を0でクリアする。

【0024】ステップ108：X座標を次に進めるために、幅テーブルのカウンタ（W）とX座標のカウンタ（X）をインクリメントする。

ステップ109：102～108までの処理を、Xがバーコードイメージ領域の右端の座標に達するまで繰り返す。

ステップ110：あるラスタ内のバーおよびスペースの幅をそれぞれの幅テーブル（TB、TS）に集計したら、次のラスタの走査を行うために、ラスタカウンタをインクリメントする。

【0025】ステップ111：バーとスペースの本数が、予め設定されている幅テーブルを完成するに十分な本数となっているか調べる。十分であれば後述する図11の102へ進み、十分でなければ112へ進む。

ステップ112：ラスタカウンタがバーコードイメージ領域の下端に達したか判定し、達していればバーとスペースの合計本数不足のため異常終了させ、達していなければ図8の102へ戻り、102～111までの処理を繰り返す。

【0026】上述した幅テーブルの作成を多数本のラスタについて行くと、統計母数は膨大になり、図5で説明した中間の分布Bbpがさらに顕著になり、それ単独で1つの分布の山を形成するに至る。図10は多数本のラスタ走査の結果擬似の分布の山ができることを示す図である。図10の（A）において、設定度数tを超える山が3つ（bp、Bbp、BP）できてしまう。したがって、ラベリングの対象が3つになって1つ余分になる。

【0027】そこで、図10の（C）のように、3つのラベリング区間のうち（（B）参照）、長い2区間をもって、細太を特定するためのラベリングを行う。図11は細太のラベリングを行う工程を示すフローチャート（その1）、図12は細太のラベリングを行う工程を示すフローチャート（その2）であり、図10に示したように3つの山ができてしまったときにも対応できるようになっている。

【0028】ステップ101：上記フローチャート1（図8および9）で作成したバーおよびスペースの幅テ

ーブルの、バーまたはスペースの合計本数が度数分布を調査するのに十分か判断する。十分でない場合は、バーまたはスペースの合計本数不足で異常終了する。なお、これは図9の111と全く同じである。

ステップ102：幅テーブルをアクセスするための、幅を表す要素変数(W)を1で初期化する。これにより左端の要素11が指定される。

【0029】ステップ103：各要素毎に合計した検出度数をストアするレジスタ(幅テーブル)の内、要素変数Wの幅テーブル T_W にストアされた検出度数がバーまたはスペースの合計本数に対して、規定の割合(α)

(図10の(A)のt参照)以上存在するかを判断する。 α は、ノイズによるバーまたはスペースの幅の変化を抑えられる値(1~3%程度)にする。幅テーブル T_W 内の値がバーまたはスペースの合計本数より大きいときは、104へ進む。 T_W 内の値がバーまたはスペースの合計本数より小さいときは、105へ進む。

【0030】ステップ104： T_W に、その幅がノイズではなく有効な幅であることを示すためのマークを書き込む。

ステップ105： T_W に、その幅がノイズによる無効な幅であることを示すためのマークを書き込む。

ステップ106：バーまたはスペースの幅テーブルの次の要素について分布状況の調査をするために、Wをインクリメントする。

【0031】ステップ107：バーまたはスペースの幅テーブルのすべての要素11について、分布状況の調査をするまで繰り返す。

ステップ108：101~107にて分布状況の調査をした結果、幅テーブル内に有効な幅であることを示すマークを書き込んだかを判断する。有効な幅であることを示すマークを書き込んでいない場合は、バーまたはスペースの分布状況異常で終了する(異常終了)。また、マークを書き込んである場合には、109へ進む。

【0032】ステップ109：幅テーブル内の有効な幅を示すマークが連続している区間が、3区間以上ある場合(図10の(B)参照)には、細太を特定するためにそのうちの2区間を限定する必要がある。そのために、有効な幅を示すマークが連続している区間毎に、区間の始点と長さを求める。

ステップ110：109で求めた各区間の長さをもとに、長い2区間を残して、それより短い区間を無効な幅であることを示すマークに置き換える。

【0033】以上の説明は、図2に示したバーコードイメージ中に何の欠陥もないことを前提にしていた。つまり、バーの一部に欠けがあったり、スペースの中に黒ドット(ゴミ)が入っていたりすることのないことを前提にしていた。しかしながら、実際にはそのような欠けやゴミの存在は必ずある。そしてこれらの欠けやゴミの量が許容範囲を超えると、バーコードの正しい認識ができ

なくなる。

【0034】このための対策として、次のような手法を採用する。すなわち、図1に示した前記第2ステップS2において、同一のイメージに対して複数回、ラスタ位置を変えながら走査を行い、各ラスタ(R1, R2...)毎に、バーおよびスペースに関するそれぞれの検出度数の合計が、該ラスタにおけるバーおよびスペースのそれぞれについての既知の一定本数と一致したことを条件に、検出度数の統計をとり幅テーブルTを生成する。

【0035】もう少し具体的に言えば、読み取り対象のバーコードの桁数が予め分かっている場合、および、ほぼ確定的な場合には、既述した読取りに対しさらに改良を加えることができる。例えば、(バーコードの桁数)×(1桁のバーまたはスペースの本数)と、ラスタから検出したバーまたはスペースの本数とが一致したときのみ、幅テーブルに検出度数を加算するようにすれば、ノイズによる幅テーブルの乱れを抑えることができる。なお、バーコードイメージの桁数と、予期していた桁数とが異なる場合には幅テーブルに対し検出度数が加算されないで、リトライ処理としてバーおよびスペースの本数に関わらず、幅テーブルに加算する処理を設けるようにしてもよい。

【0036】バーコードの桁数を図2のバーコードについて見ると、このバーコードは3桁からなる。各1桁は連続4本のバー(細太)と、これらに挟まれる連続4本のスペース(細太)とからなる。図13、図14および図15は幅テーブルを一層高精度に生成するためのフローチャートを示す。

【0037】ステップ101：複数のラスタを走査するため、ラスタカウンタを0で初期化する。ラスタカウンタは、バーコードイメージ領域内のラスタ番号を表す。また、バーおよびスペースの合計本数のカウンタ(B, S)を0で初期化する。

ステップ102：ラスタを、最も左端にある黒点を D_1 とするドット列 D_n と考え、 D_0 を1で初期化する。また、ドット列 D_n のnを特定するX座標の変数Xを1で初期化する。更に、幅テーブルの要素番号Wを1で初期化する。

【0038】ステップ103：ラスタ内のバーおよびスペースの幅の分布を格納するためのテーブル(T_B' , T_S')の各要素11を0で初期化する。また、ラスタ内のバーおよびスペースの本数のカウンタ(B' , S')を0で初期化する。

ステップ104：ドット列内の点 D_X と D_{X-1} を比較して異なっているならば、105へ進む。同じならば、109へ進む。

【0039】なお、 T_B' や T_S' のようにダッシュを付したテーブルは、検出度数を一時的にセーブするためのものである。

ステップ105：ドット列内の点 D_X と D_{X-1} が異なっ

ているならば、 D_{X-1} がバーまたはスペースの端点となるので、そのどちらか（白→黒、黒→白）判断をする。 D_X が0ならば、106へ進む。 D_X が1ならば、107へ進む。

【0040】ステップ106： D_{X-1} までがバーなので、ラスタ内のバーの幅テーブル（ $T_{B'}$ ）の内、現在カウント中の幅（WD）の要素番号W（図6の11に相当）の要素 $T_{B' W}$ をインクリメントする。また、バーを新たに検出したので、ラスタ内のバーの合計本数のカウンタ（ B' ）（図6の12に相当）をインクリメントする。

【0041】ステップ107： D_{X-1} までがスペースなので、ラスタ内のスペースの幅テーブル（ $T_{S'}$ ）の内、現在カウント中の幅（WD）の要素番号Wの要素 $T_{S' W}$ をインクリメントする。また、スペースを新たに検出したので、ラスタ内のスペースの合計本数のカウンタ（ S' ）（図6の12に相当）をインクリメントする。

【0042】ステップ108：バーまたはスペースの検出をしたので、次のスペースまたはバーの幅をカウントするために、幅テーブルカウンタ（W）を0でクリアする。

ステップ109：X座標（図7のX）を次に進めるために、幅テーブルのカウンタ（W）とX座標のカウンタ（X）をインクリメントする。

ステップ110：102～108までの処理を、X座標がバーコードイメージ領域の右端に達するまで繰り返す。

【0043】ステップ111：ラスタ内のバーとスペースの合計本数（ B' 、 S' ）と、仮定した桁数でのバーとスペースの合計本数（例えば、図2に示したバーコードの場合には、各桁が8本のバーとスペースで構成されているので、合計本数は桁数 $\times 8 - 1$ となる。）とが等しければ、ステップ112へ進む。異なれば、113へ進む。リトライ後の場合は、常に112へ進む。

【0044】ステップ112：バーおよびスペースの本数カウンタ（ B' 、 S' ）に、ラスタ内で検出したそれぞれの本数（ B' 、 S' ）を加算する。

ステップ113：あるラスタ内のバーおよびスペースの幅をそれぞれ幅テーブルに集計したら、次のラスタの走査を行うために、ラスタカウンタをインクリメントする。

【0045】ステップ114：ラスタカウンタがバーコードイメージ領域の下端に達するまで、繰り返す。

ステップ115：バーとスペースの本数が、予め設定されている幅テーブルを完成するに十分な本数となっているか調べる。十分であれば、図11の102へ進み、十分でなければ図13の101に戻る。

【0046】図16はラベリングの別の手法を説明するための図である。前述した図5や図10に示すラベリン

グは、幅テーブルを参照してこれを一定のしきい値でスライスすることによりラベリングを行った。ところがこのしきい値の設定を誤ると、正しいラベリングができない。そこで図16の手法では検出度数の分布曲線から、ラベリングを行うこととする。図16の（A）に示す分布曲線は、同図（B）のようにU、D、およびVという変化及び、同図（C）のようにU、H、D、およびLという変化を示す。この変化の様子から同図（D）のようにラベリングが行える。

【0047】すなわち、図1に示す第3ステップS3において、幅テーブルを参照し、幅を変数とする検出度数の曲線から該検出度数の上昇（U）、下降（D）および横違い（V）を算出して、該曲線の極大および極小を割り出し、該極大を中心とし、かつ、該極小により区分された幅の範囲を切り出して、該幅の細太を特定する前記のラベリングを行う。

【0048】図17～図24は分布曲線からラベリングを行う工程を示すフローチャートである。

ステップ101：幅テーブルをアクセスするための要素番号変数（W）を1で初期化する。幅テーブルの1番目の要素 T_1 （図6の11に相当）に対する比較対象となる幅テーブルの0番目の要素 T_0 を0で初期化する。

【0049】ステップ102：幅テーブルの一つ前の要素と値の比較をする。この場合 β は、ノイズマージンを大きくするために加えるオフセットであり、度数の合計の1%程度に設定する。一つ前の要素より大きければ、104へ進む。

ステップ103：幅テーブルの一つ前の要素と値の比較をする。一つ前の要素より小さければ、105へ進む。そうでなければ、106へ進む。

【0050】ステップ104：要素番号Wに対応して設けられた増減テーブル（ C_W ）に上昇を示すラベリング（U）を行い、107へ進む。

ステップ105：増減テーブル（ C_W ）に下降を示すラベリング（D）を行い、107へ進む。

ステップ106：増減テーブル（ C_W ）に横違いを示すラベリング（V）をする。

【0051】ステップ107：Wが幅テーブルの最大要素番号（図16の（A）に示すイメージの右端）を超るまで繰り返す。

ステップ108：増減テーブルの最初の要素（ C_0 ）に下降（これから始まる上昇とは逆の下降に設定する）を示すラベリングを、最後の要素（ C_{MAX} ）にこれまでの下降とは逆の上昇を示すラベリングをする。これは、次に述べる極大極小のラベリングをする処理を簡単にするための前処理である。

【0052】ステップ109：101～107でラベリングした増減テーブルに、さらに極大極小のラベリングをするためもう一度増減テーブルをアクセスする。そこで再びWを0で初期化する。

ステップ110: 増減テーブルの一つ後の要素とラベリングの比較をする。ラベリングが異なる場合は、111へ進む。ラベリングが同一の場合は、124へ進む。

【0053】ステップ111: 一つ後のラベリングが上昇の場合は、横違いから上昇、あるいは、下降から上昇であるが、108の処理と113~119の処理により、横違いから上昇のパターンではこの処理に入らないので、後者の下降から上昇のパターンが発生するのみである。この場合は、122へ進む。

ステップ112: 一つ後のラベリングが下降の場合は、横違いから下降、あるいは、上昇から下降であるが、108の処理と113~119の処理により、横違いから下降のパターンではこの処理に入らないので、後者的上昇から下降のパターンが発生するのみである。この場合は、123へ進む。

【0054】ステップ113: 一つ後が横違いの場合は、さらにその後のラベリングを調査するために、別のアクセス用の変数(i)を使用する。iは2つ先をアクセスさせるために(1つ先は分かっている)ので2で初期化する。

ステップ114: 増減テーブル(C_{W+i})のラベリングが横違いの場合には、115へ進み、横違い以外ならば、117へ進む。

【0055】ステップ115: iをインクリメントする。

ステップ116: i+Wが増減テーブルの最大要素番号を超えるまで、繰り返す。

ステップ117: 増減テーブル(C_W)と(C_{W+i})のラベリングが異なれば、118へ進む。これらのラベリングが同一ならば、121へ進む。

【0056】ステップ118: 増減テーブル(C_W)が上昇ならば、120へ進む。(C_W)が下降ならば、119へ進む。

ステップ119: ラベリングが横違いから上昇に転じる直前(C_{W+i-1})まで、極小のラベリングを行い121へ進む。

ステップ120: ラベリングが横違いから下降に転じる直前(C_{W+i-1})まで、極大のラベリングをする。

【0057】ステップ121: 横違いの区間の処理は終わったので、Wにiを加える。

ステップ122: 下降から上昇に転じる点は極小なので、増減テーブル(C_W)に極小のラベリングを行い、124へ進む。

ステップ123: 上昇から下降に転じる点は極大なので、増減テーブル(C_W)に極大のラベリングを行い、124へ進む。

【0058】ステップ124: 増減テーブルの次の要素を処理するために、Wをインクリメントする。

ステップ125: 増減テーブルの最大要素番号-1を超えるまで繰り返す。

ステップ126: 細太を分布の極小点で区切るために、増減テーブルを参照しながら、対応する幅テーブルにラベリングをする。そのために、Wを1で初期化する。細太の分布比率をラベリングの参考にするために、度数累計用変数(S)を0で初期化する。また、バーまたはスペースの幅の種類数を示すラベリング値を1で初期化する。

【0059】ステップ127: Sに幅テーブル(T_W)の度数を加える。

ステップ128: 増減テーブル(C_W)が極大かつ、増減テーブルの一つ前(C_{W-})から極小が連続していなければ、129へ進む。

ステップ129: 幅テーブルの区間($T_1 \sim T_W$)の合計(S)が、細太の区別を付ける基準として予め設定されている全体の度数に対する細いバーまたはスペースの比率の内、最も小さい値と全体の度数とを掛けた値(θ)と、最も大きい値と全体の度数とを掛けた値(δ)との間にあれば、130へ進む。そうでなければ、131へ進む。

【0060】ステップ130: 細太の区別を付けるために、ラベリング値をインクリメントする。

ステップ131: 幅テーブル(T_W)に度数が存在するならば、132へ進む。存在しないならば、133へ進む。

ステップ132: 幅テーブル(T_W)にラベリング値を書き込む。

【0061】ステップ133: 幅テーブル(T_W)に無効な幅を示すラベリング値を書き込む。

ステップ134: Wが幅テーブルの最大要素番号に達するまで127~133までの処理を繰り返す。

ステップ135: ラベリング値がバーまたはスペースの幅の種類よりも多いかまたは少ないときは、幅テーブルの分布が異常であるものとして終了する。等しいときは、正常に幅テーブルのラベリングが終了する。

【0062】尚、本実施例ではバーとスペース各々に対して幅テーブルのラベリングを行っているが、バーあるいはスペースの幅だけで情報認識が可能なバーコードパターン場合には幅テーブルのラベリングはバーあるいはスペースのどちらか一方だけでよい。また、本実施例では、バーコードイメージに対して複数回ラスタの位置を変えながら走査を行っているが、ノイズの少ないバーコードイメージに対しては一回の走査でもよい。

【0063】

【発明の効果】以上説明したように本発明によれば、認識対象のバーコードイメージそのものからバーおよびスペースの細太判断基準をまず作り、これを基準にして細太を決定するから認識率はきわめて高くなる。また従来のモジュール幅の指示を行う手法に比べると操作はきわめて簡素化される。

【0064】また、バーコードイメージにノイズがの

り、イメージが乱れた場合にも乱れたイメージでの細太の基準を作り出すので、認識率が数パーセント向上する。特に、解像度の低いスキナのイメージを偏倍により擬似的に解像度を高めるといことがしばしば行われるが、このような場合のパーおよびスペースの幅のパラツキに対して効果が高い。この場合、スキナのコストを削減できるメリットもある。

【図面の簡単な説明】

【図1】本発明に係る方法の基本ステップを表すフローチャートである。

【図2】バーコードパターンの周知の一例を表す図である。

【図3】幅テーブルを作成するための統計処理を表す図である。

【図4】各幅の細太を特定するラベリングを説明する図である。

【図5】隣接する分布が重なる場合の統計処理を説明する図である。

【図6】幅テーブルを作成するための統計処理を具体例で示す図である。

【図7】図2における任意の1つのラスタにおけるバーコードイメージの一部を示す図である。

【図8】幅テーブルを作成する工程を示すフローチャート（その1）である。

【図9】幅テーブルを作成する工程を示すフローチャート（その2）である。

【図10】多数本のラスタ走査の結果擬似的分布の山ができることを示す図である。

【図11】細太のラベリングを行う工程を示すフローチャート（その1）である。

【図12】細太のラベリングを行う工程を示すフローチャート（その2）である。

ャート（その2）である。

【図13】一層高精度に幅テーブルを生成するためのフローチャート（その1）である。

【図14】一層高精度に幅テーブルを作成するためのフローチャート（その2）である。

【図15】一層高精度に幅テーブルを作成するためのフローチャート（その3）である。

【図16】ラベリングの別の手法を説明するための図である。

【図17】分布曲線からラベリングを行う工程を示すフローチャート（その1）である。

【図18】分布曲線からラベリングを行う工程を示すフローチャート（その2）である。

【図19】分布曲線からラベリングを行う工程を示すフローチャート（その3）である。

【図20】分布曲線からラベリングを行う工程を示すフローチャート（その4）である。

【図21】分布曲線からラベリングを行う工程を示すフローチャート（その5）である。

【図22】分布曲線からラベリングを行う工程を示すフローチャート（その6）である。

【図23】分布曲線からラベリングを行う工程を示すフローチャート（その7）である。

【図24】分布曲線からラベリングを行う工程を示すフローチャート（その8）である。

【符号の説明】

11…要素

12…積算カウンタ

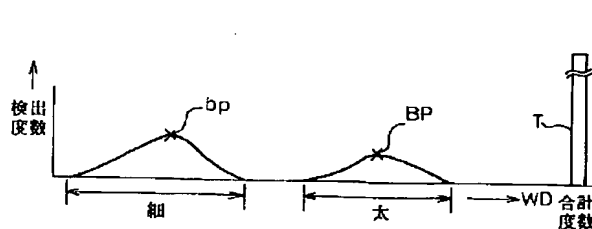
R1, R2, R3…ラスタ

BP, bp…バー

SP, sp…スペース

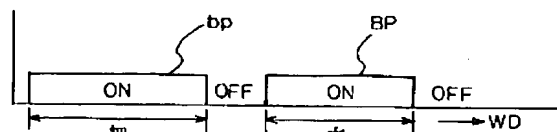
【図3】

幅テーブルを作成するための統計処理を表す図



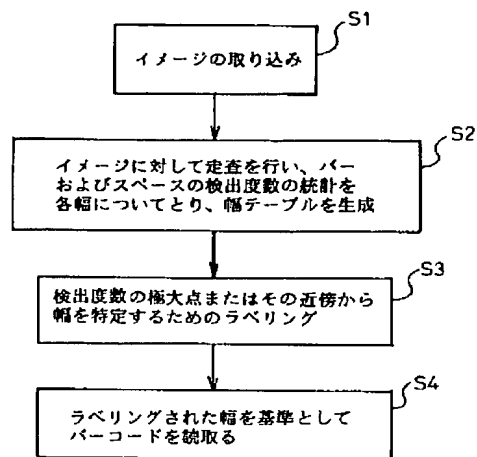
【図4】

各幅を細太を特定するラベリングを説明する図



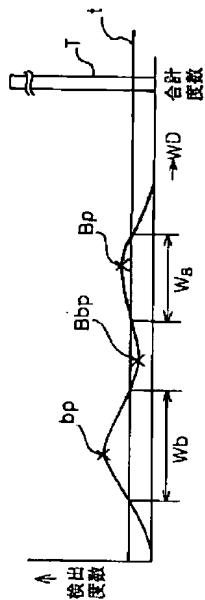
【図1】

本発明に係る方法の基本ステップを表すフローチャート



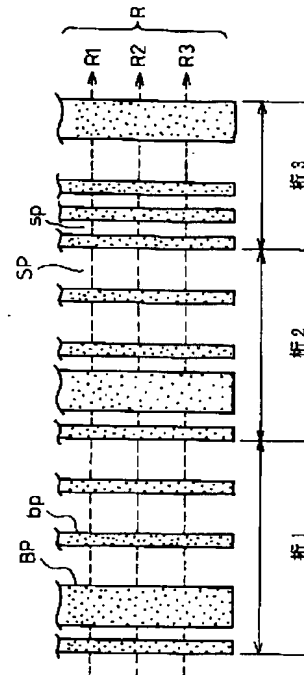
【図5】

隣接する分布が重なる場合の統計処理を説明する図



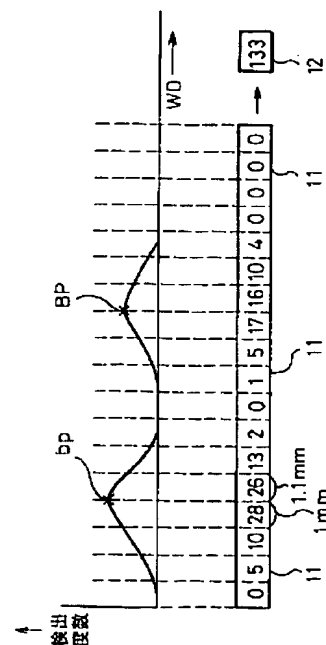
【図2】

バーコードパターンの周知の一例を表す図



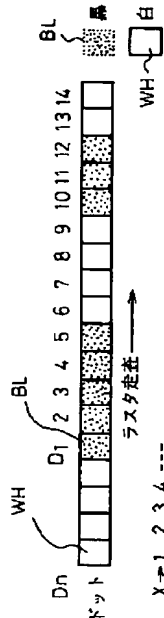
【図6】

幅テーブルを作成するための統計処理を具体例で示す図



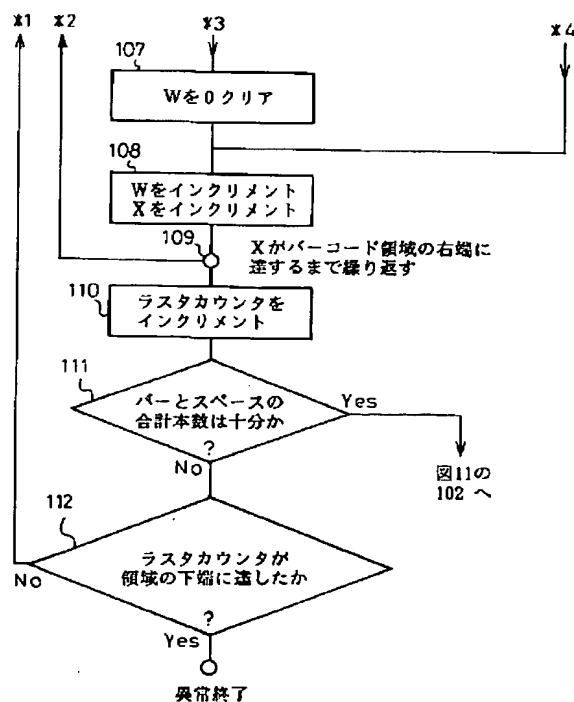
【図7】

図2における任意の1つのラスタにおけるバーコードイメージの一部を示す図



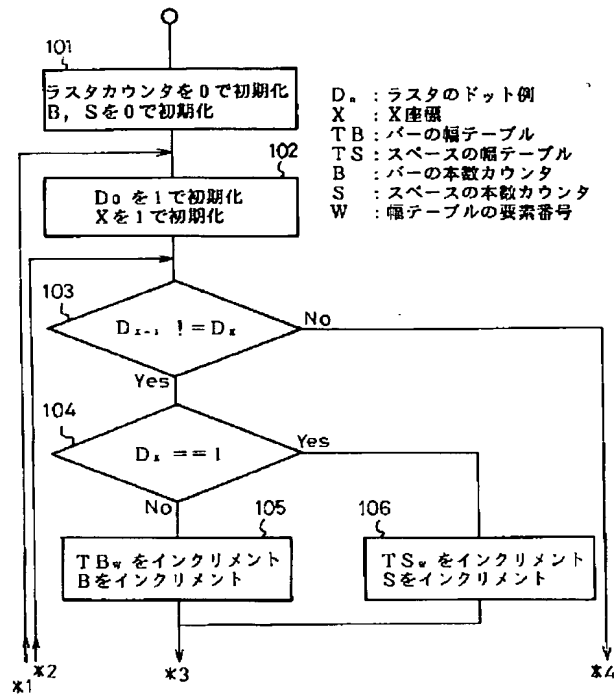
【図9】

幅テーブルを作成する工程を示すフローチャート（その2）



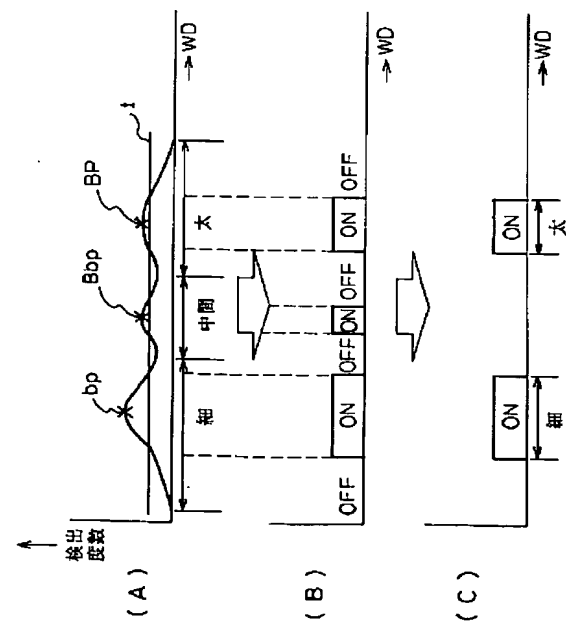
【図8】

幅テーブルを作成する工程を示すフローチャート（その1）



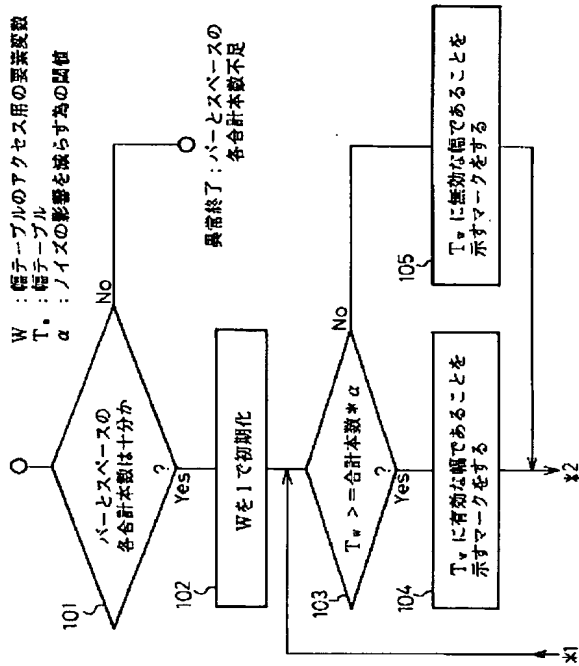
【図10】

多数本のラスタ走査の結果類似の分布の山ができることを示す図



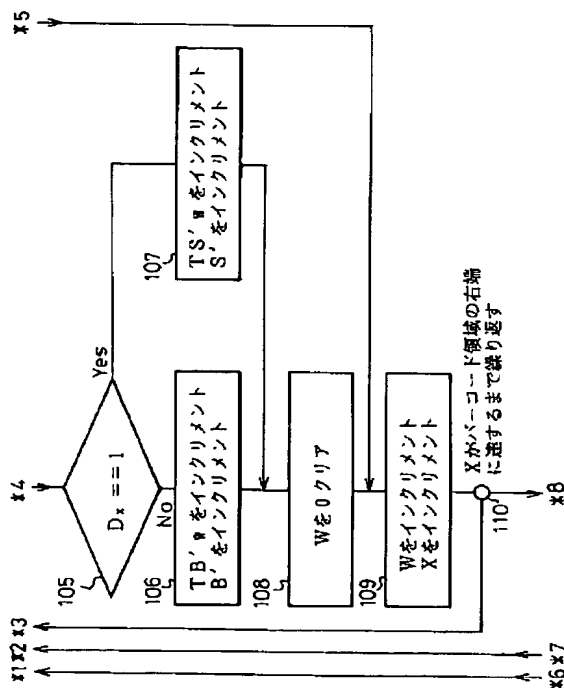
【図11】

細大のラベリングを行う工程を示すフローチャート（その1）



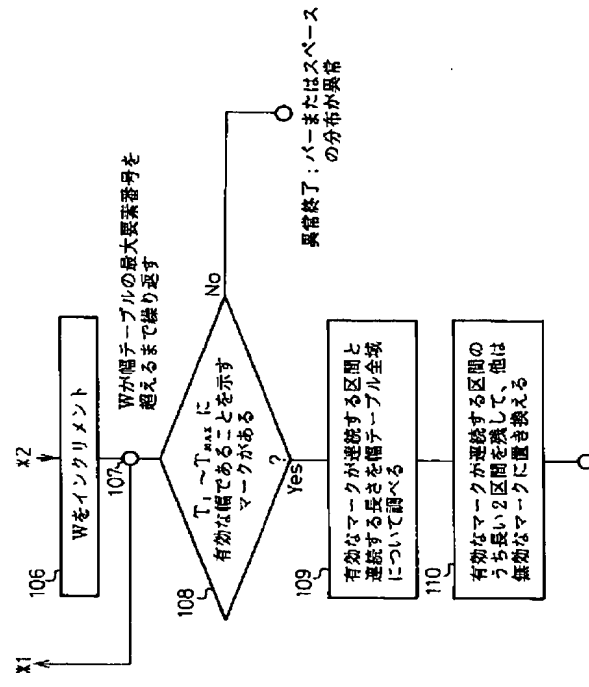
【図14】

一画素精度に幅テーブルを生成するためのフローチャート（その2）



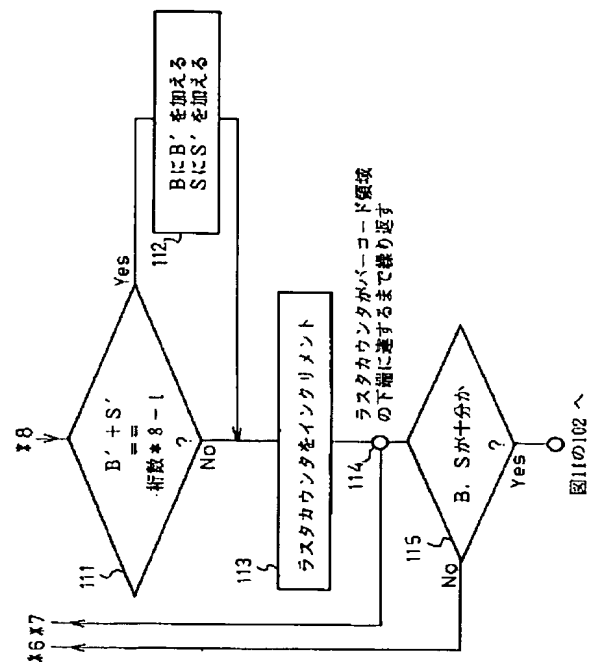
【図12】

細大のラベリングを行う工程を示すフローチャート（その2）



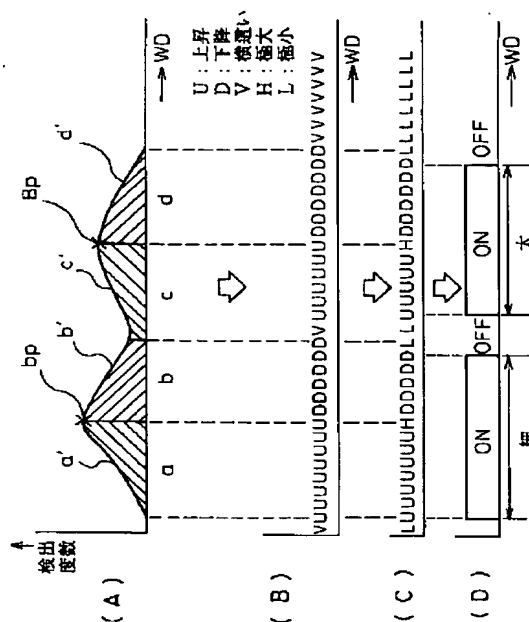
【図15】

一画素精度に幅テーブルを生成するためのフローチャート（その3）



【图 16】

ラベリングの別の手法を説明するための図



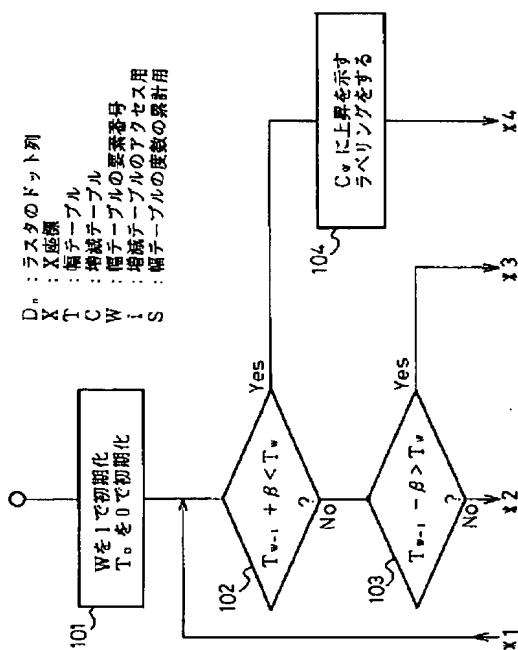
【图 18】

分布曲線からラベリングを行う工程を示すフローチャート（その1）

```

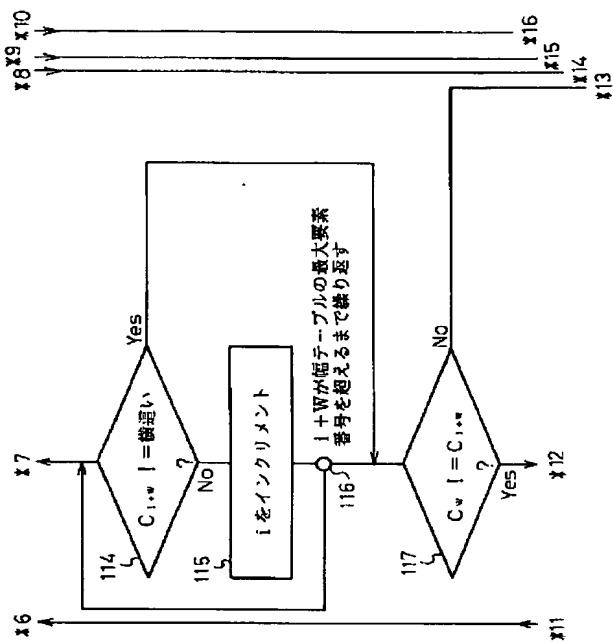
graph TD
    x1((x1)) --> 106[106]
    x2((x2)) --> 106
    x3((x3)) --> 105[105]
    x4((x4)) --> 105
    106 --> 107((107))
    105 --> 108((108))
    108 --> 109[109]
    109 --> 110[110]
    110 --> x5((x5))
    x5 --> 108
    108 --> 107
    107 --> 105
    107 --> 109
    
```

Figure 1 is a flowchart illustrating a neural network processing method. The process starts with inputs x_1 , x_2 , x_3 , and x_4 . x_1 and x_2 are fed into block 106, which outputs to a summing junction 107. x_3 and x_4 are fed into block 105, which outputs to a summing junction 108. Block 108 outputs to block 109, which outputs to block 110. Block 110 outputs to x_5 . A feedback loop from x_5 goes through block 108 and back to block 107. A decision diamond 107 checks if W exceeds the maximum value. If yes, it loops back to block 105. If no, it goes to block 109.



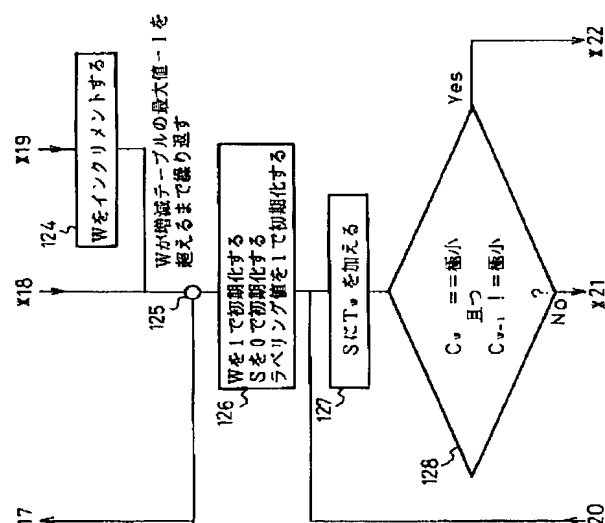
【圖 20】

分布曲線からラベリングを行う工程を示すフローチャート（その4）



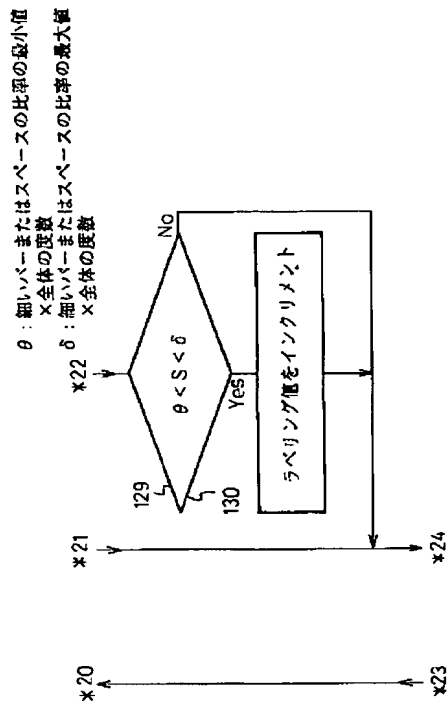
【圖 22】

分布曲線からラベリングを行う工程を示すフローチャート（その8）



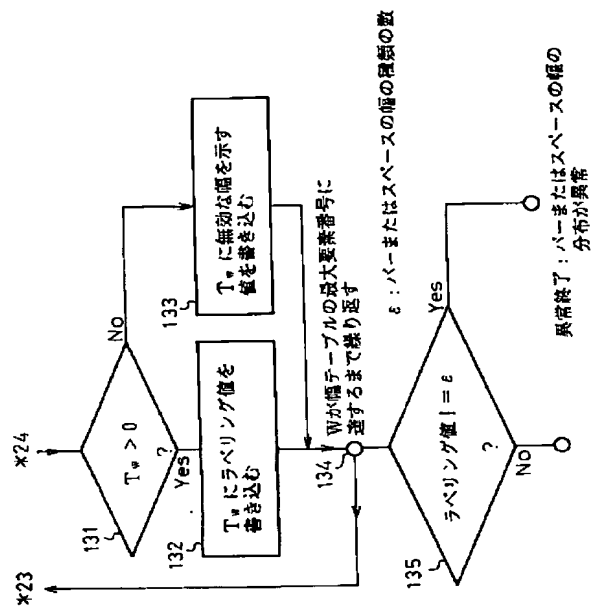
【図23】

分布曲線からラベリングを行う工程を示すフローチャート（その7）



【図24】

分布曲線からラベリングを行う工程を示すフローチャート（その8）



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☒ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☒ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.